Control Engineering Practice 125 (2022) 105211

Contents lists available at ScienceDirect

Control Engineering Practice

journal homepage: www.elsevier.com/locate/conengprac

Positioning of aircraft relative to unknown runway with delayed image data, airdata and inertial measurement fusion

Tamás Gróf [a], Péter Bauer [a,b,*], Yoko Watanabe [c]

[a] Systems and Control Laboratory, Institute for Computer Science and Control, ELKH Kende utca 13.-17., Budapest H-1111, Hungary
[b] Department of Control for Transportation and Vehicle Systems, Budapest University of Technology and Economics, Stoczek u. 2., Budapest H-1111, Hungary
[c] Department of Information Processing and Systems, ONERA - The French Aerospace Laboratory, 2 avenue Edouard Belin, 31000 Toulouse, France

ARTICLE INFO

Keywords:
Nonlinear estimation
Sensor fusion
Aircraft landing
Machine vision

ABSTRACT

This work addresses the challenge of providing precise runway-relative position, velocity and orientation reference to a landing aircraft based on monocular camera and low-cost inertial sensor data complemented by barometric sensor readings. GPS information is excluded from this sensor set because it is intended to use the developed estimator for GPS and Instrumental Landing System fault detection. The characteristic size of the runway is assumed to be unknown and it is estimated run-time after verifying global observability of the nonlinear system. The delay caused by image processing is dealt with a delayed-Error-State Kalman Filter (ESKF). This algorithm considers dynamic propagation of the image information between acquisition and application forward in time thus the delay does not appear in the system dynamics. The first evaluation of the estimator is done for ideal simulated data to verify applicability of the delayed-ESKF and its flawless implementation. Then more realistic simulated data with sensor biases and noise is considered to verify closer to realistic performance and bias estimation precision. Finally, the estimator is tested with real flight data collected in the VISION EU H2020 research project. Estimation results are compared to GPS SBAS-based data and Airbus precision tolerances showing satisfactory performance. The methodological contribution of the paper is the unique combination of existing methods and ideas leading to a new solution proven to work satisfactorily even with real flight data.

1. Introduction

In recent years several projects aimed to provide analytical redundancy (Goupil et al., 2015) and additional information sources to on-board aircraft systems. Camera sensors are getting considered as an additional source of information as they are becoming more popular not only on unmanned aerial vehicles (UAVs) but also on passenger airplanes (Gibert et al., 2018). A Europe–Japan collaborative research project called VISION (Validation of Integrated Safety-enhanced Intelligent flight cONtrol) has explored the possibilities to use camera systems during aircraft landing between 2016–2019 see VISION (2016). VISION focused on critical flight scenarios especially on near ground maneuvers and aimed to improve the overall precision level of current navigation systems by adding image information. The work presented here is part of the project. It focuses on IMU (Inertial Measurement Unit) and camera-based runway-relative positioning of the aircraft excluding GPS (Global Positioning System) information as the results are applied in GPS and Instrumental Landing System (ILS) fault detection (see Grof et al. (2019) and Grof and Bauer (2021)).

Exploring the literature of camera-based runway-relative positioning and navigation there are mainly two sets of possible approaches. The first set called visual servoing considers landing the aircraft based-on the observed visual features without explicitly estimating its position and velocity relative to the runway see e.g. Azinheira and Rives (2008), Le Bras et al. (2009), Gibert and Puyou (2013) and Burlion and Kolmanovsky (2020).

The second set of methods considers position estimation relative to the surroundings. The work presented in this paper falls into this set. Table 1 summarizes the main contributions of the most relevant existing literature sources (including also the author's preceding work (Grof et al., 2019)) regarding the applied sensors and the covered estimates and challenges. The first column shows the contributions of the current work for comparison. The first six rows contain the available sensor units that are frequently used in aerospace applications. The second six rows include the possible states to be estimated and other challenges to be handled. In the next paragraph only the main characteristics of the referenced articles will be mentioned. Further literature sources related to this approach are e.g. Andert and Mejias (2015), Conte and Doherty (2009) and Martinelli (2011).

In Zhang et al. (2019) an infrared-inertial navigation system with barometric unit and radio altimeter is proposed for aircraft landing. It

* Corresponding author at: Systems and Control Laboratory, Institute for Computer Science and Control, ELKH Kende utca 13.-17., Budapest H-1111, Hungary.
E-mail addresses: groftamas96@gmail.com (T. Gróf), bauer.peter@sztaki.hu (P. Bauer), Yoko.Watanabe@onera.fr (Y. Watanabe).

https://doi.org/10.1016/j.conengprac.2022.105211
Received 15 September 2020; Received in revised form 4 May 2022; Accepted 5 May 2022
Available online 20 May 2022

Table 1
Contributions relative to existing literature and previous work of the authors.

	Current work	Zhang et al. (2019)	Watanabe et al. (2019)	Watanabe et al. (2020)	Gibert et al. (2018)	Hiba et al. (2021)	Lynen et al. (2013)	Joo et al. (2008)	Weiss et al. (2013)	Strydom et al. (2014)	Huang et al. (2017)	Grof et al. (2019)
GPS		x	x	x			x	x				
IMU	x	x	x	x	x		x	x	x		x	x
Baro	x	x	x	x			x					
Mono vision	x	x	x		x	x	x	x	x		x	x
Stereo vision				x						x		
Radio altimeter		x										
Position	x	x	x	x	x	x	x	x	x	x	x	x
Velocity	x	x	x	x	x		x	x	x		x	x
Orientation	x	x	x	x	x	x	x	x	x		x	x
IMU Biases	x	x	x	x			x		x			x
Delay handling	x		x	x			x	x				
Unknown runway	x				x							

considers geographic terrain information and presents thorough comparison with other existing approaches for demonstrating the superior performance of the developed algorithm in terms of accuracy.

Watanabe et al. (2019) considers vision-based aircraft relative navigation to a runway with known size (in frame of the VISION project) and applies a delayed-ESKF which utilizes the image trigger signal to compensate the image processing delay forward in time.

Watanabe et al. (2020) extends the previous method by incorporating fault detection and protection level calculation. The developed algorithm was implemented onboard the VISION test aircraft and real flight tested.

Gibert et al. (2018) addresses the vision-based estimation of runway-relative position assuming that the IMU provides ground relative velocity and orientation with sufficient precision (industrial grade sensors) and the runway sizes are unknown.

Hiba et al. (2021) presents a method to estimate runway-relative parameters solely from monocular camera images with known runway sizes.

Lynen et al. (2013) proposes a ring-buffer scheme to process multiple delayed measurements. The proposed Multi-Sensor-Fusion EKF (MSF-EKF) is tested in outdoor navigation with a combined Simultaneous Localization and Mapping (SLAM) system.

In Joo et al. (2008) an EKF (Extended Kalman Filter) formula with delayed vision measurements is developed considering the noise correlation during landing scenarios. The aircraft is equipped with GPS and IMU unit while a moving rover along the runway monitors the aircraft with a monocular camera. Bearing observations and rover position are sent through wireless communication (causing most of the delay) and then the onboard and external sources are fused with the proposed delayed-EKF algorithm.

Weiss et al. (2013) introduces a loosely-coupled IMU-Camera navigation filter in GPS (Global Positioning System) denied environment. It estimates both acceleration and angular rate sensor biases. Nonlinear observability analysis is performed and real flight test results underline the applicability of the method requiring hundreds of reference points to work properly.

Strydom et al. (2014) applies stereo vision and optic flow to position a quadcopter along a trajectory by tracking several hundred (400 in the example) feature points. The method is limited to 0–15 m flight altitude range because of the stereo vision.

Huang et al. (2017) proposes a concept for UAV localization by applying apriori known landmarks but it ignores the possible measurement biases of the inertial unit.

In a previous work of the authors (Grof et al., 2019) a monocular vision-based landing aircraft state estimation was presented neglecting image processing delay and assuming known runway sizes. The method considered only three reference points related to the runway which is much less then the several points considered in other works.

To sum up the above part of literature survey one of the main challenges is to estimate IMU sensor bias and unknown runway size at the same time with low-cost IMU and without any apriori knowledge (such as geo-localized landmarks). The common challenges which are only partially covered by each source can be listed as (see also Table 1):

- The requirement to detect and track a large set of points in the image or to use geo-referenced images
- The lack of acceleration and/or angular rate sensor bias estimation which is unavoidable with low-cost IMU sensory systems
- The lack of precisely known velocity and/or orientation from other systems
- The consideration of delay caused by image processing
- The assumption of known runway geometry

The goal of the current work is to fuse ideas and concepts from the existing literature in a unique way applying measurements from a limited set of sensor types and providing all required estimates. The applied sensory measurements are IMU acceleration and angular rate, barometer readings and mono camera runway corner points and vanishing point readings. GPS is excluded from this sensor set as it is intended to use the solution for GPS fault detection. The targeted states to be estimated are runway-relative position, velocity and orientation together with acceleration and angular rate biases and unknown runway size. Table 1 shows that none of the published methods can cover such a large range of parameters especially based-on low-cost IMU data (e.g. in Gibert et al. (2018) high precision aerospace grade IMU data is considered). The effects of the image processing delay are considered by implementing a delayed-ESKF algorithm (Watanabe et al., 2019) knowing that the delay is measurable on the VISION test aircraft.

The developed method is first evaluated with simulated ideal data (no noise, no biases), then with simulated realistic data (noise and sensor biases). Finally, the method is applied and tested on real flight data which was collected in frame of the VISION project to prove real-life applicability.

The main contributions of this paper relative to the literature are listed below:

- Considering unknown runway width with low-cost IMU and the addition of barometric measurements. The fact that the knowledge of the runway parameters is not required can be essential during emergency landings of small UAVs
- Considering image processing delay with delayed-ESKF algorithm
- Verification of the delayed-ESKF solution with ideal (error/noise free) simulated data

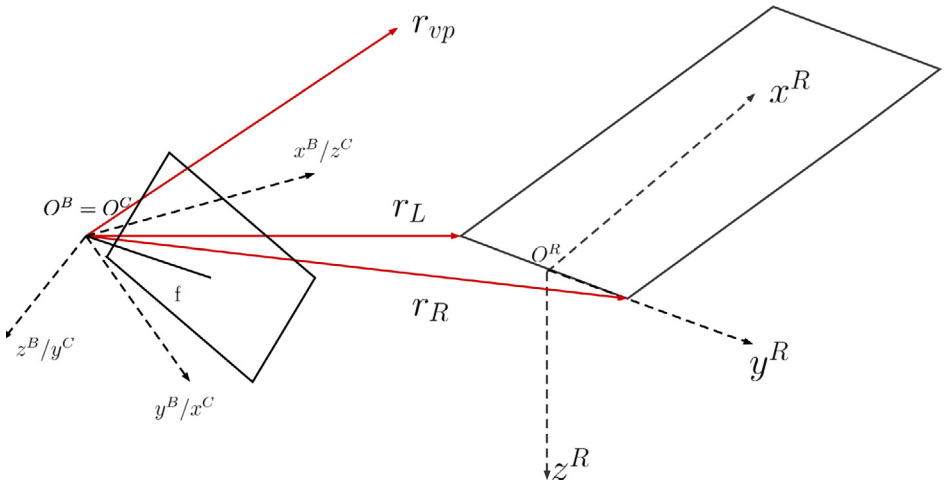

Fig. 1. Reference systems for the simulated data.

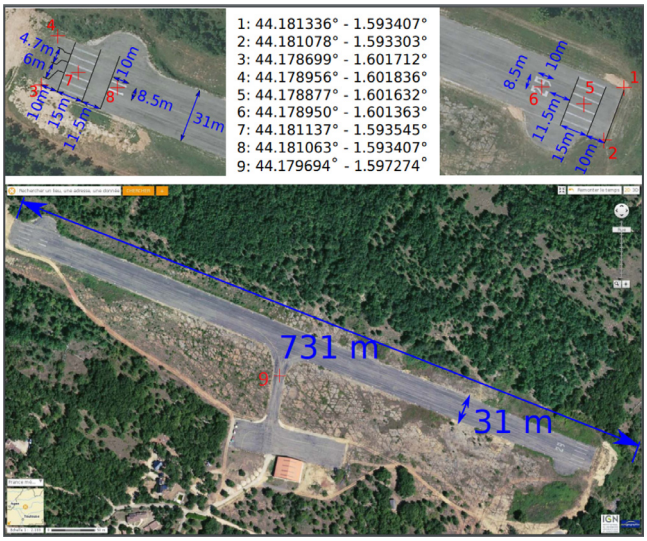

Fig. 2. Septfonds (France) runway characteristics and significant LLA coordinates.

Fig. 3. Camera positions under the wings of the K-50 test aircraft.

- Verification of the whole algorithm with real flight UAV data from VISION project and evaluation of its precision in position and orientation considering industrial tolerances provided by Airbus

The structure of the article is as follows: in Section 2 the related mathematical model is presented. Section 3 analyzes the observability of the system. Next, in Section 4 the proposed delayed-ESKF estimator algorithm is introduced. Then Sections 5 and 6 validate the proposed navigation system both with ideal and realistic simulated data. In Section 7 the evaluation of the solution with real flight data including delayed image information is presented. Finally, Section 8 concludes the paper.

2. Runway-relative dynamics and measurements

This section summarizes the state dynamic and measurement equations applied in the formulation of the state estimator and gives some information about the test aircraft and airfield.

In any flight dynamics related problem one has to first define the applied reference systems. These are the North-East-Down (NED) frame (X^E, Y^E, Z^E) considered to be an inertial one because of the short flight distances during landing, the fixed runway frame (X^R, Y^R, Z^R) with origin (O^R) placed at the center of threshold bars, the body frame (X^B, Y^B, Z^B) which is fixed to the aircraft and lastly the camera frame (X^C, Y^C, Z^C) which can be body system aligned as in Fig. 1 or unaligned as in Fig. 4. In case of real flight the camera was located under one of the wings of the K-50 test aircraft meaning that the camera frame is unaligned. This is shown in Figs. 3 and 4.

Fig. 2 shows the runway (Septfonds, France) on which the real flight test was executed. The Latitude–Longitude–Altitude (LLA) of the origin of the NED frame is assigned at $\begin{bmatrix}-44.179694° & 1.597274° & 193\ \text{m}\end{bmatrix}$ denoted as *No. 9*. This is the point of aircraft preparation before taxi and take-off. The NED coordinates of the origin O^R of the runway frame (see Fig. 1) were $\begin{bmatrix}164.16\ \text{m} & -306.265\ \text{m} & 6.72\ \text{m}\end{bmatrix}$.

After defining the reference systems the mathematical model is formulated from the kinematic equations including the following variables (similar to Watanabe et al. (2019)):

$$x = \begin{bmatrix} p_R^T & v_R^T & q^T & b_a^T & b_\omega^T & W \end{bmatrix}^T \tag{1}$$

$$u = \begin{bmatrix} a_B^T & \omega_B^T \end{bmatrix}^T \tag{2}$$

$$\eta = \begin{bmatrix} \eta_a^T & \eta_\omega^T & \eta_{ba}^T & \eta_{b\omega}^T & \eta_W^T \end{bmatrix}^T \tag{3}$$

$$z = \begin{bmatrix} z_L^T & z_R^T & z_{vp}^T & z_{baro}^T \end{bmatrix}^T \tag{4}$$

$$\nu = \begin{bmatrix} \nu_{z_L}^T & \nu_{z_R}^T & \nu_{z_{vp}}^T & \nu_{z_{baro}}^T \end{bmatrix}^T. \tag{5}$$

Here, x is the state vector with $p_R = \begin{bmatrix} p_x & p_y & p_z \end{bmatrix}$ runway-relative position, $v_R = \begin{bmatrix} V_x & V_y & V_z \end{bmatrix}$ runway-relative velocity, $q = \begin{bmatrix} q_0 & q_1 & q_2 & q_3 \end{bmatrix}$ quaternion representation of runway-relative orientation and $b_a = \begin{bmatrix} b_{a_x} & b_{a_y} & b_{a_z} \end{bmatrix}$, $b_\omega = \begin{bmatrix} b_p & b_q & b_r \end{bmatrix}$ bias parameters of accelerometers and gyroscopes. Finally W is the runway threshold bar width added as a new state (31 m in Fig. 2). The u input vector includes the IMU measurements such as acceleration $a_B = \begin{bmatrix} a_x & a_y & a_z \end{bmatrix}$ and angular rate $\omega_B = \begin{bmatrix} p & q & r \end{bmatrix}$ in body frame. The η vector consists of the process noise variables affecting IMU measurements η_a, η_ω, IMU bias values $\eta_{ba}, \eta_{b\omega}$ and runway width state η_W. The latter three are artificial noises influencing the bias values and runway width which are modeled as first-order Markov processes. z includes the measurements with z_L and z_R image plane coordinates of left and right threshold line corners, z_{vp} projection of the vanishing point and z_{baro} barometric altitude measurement. The related measurement noise parameters in ν are ν_{z_L}, ν_{z_R}, $\nu_{z_{vp}}$ and $\nu_{z_{baro}}$. Note that the model equations are delay free similarly to Watanabe et al. (2019) as the consideration of the known image measurement delay does not require delayed dynamics. The measurement update is done considering the dynamic effects between image acquisition and update forward in time as presented in Section 4.

The kinematic equations that describe the aircraft motion are presented in Eqs. (7) to (11) in input affine form:

$$\begin{aligned} \dot{x} &= f(x,\eta) + \sum_{i=1}^{m} g_i(x) u_i \\ z &= h(x,\nu). \end{aligned} \tag{6}$$

$$\dot{p}_R = v_R = f_{p_R}(x,\eta) + 0 \tag{7}$$

$$\dot{v}_R = T_{RB}(a_B - b_a - \eta_a) + g = f_{v_R}(x,\eta) + g_{v_R}(x) a_B \tag{8}$$

$$\dot{q} = -Q(q)\omega_B + Q(q)(b_\omega + \eta_\omega) = f_q(x,\eta) + g_q(x)\omega_B \tag{9}$$

$$\dot{b} = \begin{bmatrix} \dot{b}_a \\ \dot{b}_\omega \end{bmatrix} = \begin{bmatrix} 0_{6\times6} & I_6 & 0_{6\times1} \end{bmatrix} \eta = f_b(x,\eta) + 0 \tag{10}$$

$$\dot{W} = \begin{bmatrix} 0_{1\times12} & 1 \end{bmatrix} \eta = f_W(x,\eta) + 0 \tag{11}$$

Here, $f_{v_R}(x,\eta) = T_{RB}(-b_a - \eta_a) + g$ and $g_{v_R}(x) = T_{RB}$ are the components of the input affine form of the differential equation for v_R. The other terms are formulated similarly. If there is a 0 on the right hand side that means no input effect in the given equation ($g_i(x) = 0$). Here T_{RB} is the body to runway transformation matrix and $Q(q)$ is the matrix with quaternion terms in the quaternion dynamics similarly to Weiss et al. (2013). I_6 is the six dimensional unit matrix.

The measurement equations were formulated by using a perspective camera projection model. The first two reference points are the threshold line corners of the runway while the third is the so called vanishing point aligned with the runway heading direction (coinciding with the runway frame X^R axis). Details about runway feature detection can be found in Hiba et al. (2018) and Hiba et al. (2021).

When considering camera position away from the CoG of the aircraft a Δp_{cam} position and Δq_{cam} orientation transformation of the camera (see Fig. 4) should be applied to get the correct camera position and orientation in runway frame

$$p_R^C = p_R + T_{RB}\Delta p_{cam} \tag{12}$$

$$q_R^C = q \otimes \Delta q_{cam}. \tag{13}$$

Here, $\otimes$ refers to the quaternion product as in Sola (2017). The rotation matrix from the runway frame to the camera frame can be expressed as $T(q_R^C)$. The camera frame coordinates of the threshold bar corners and the vanishing point can be obtained as

$$r_{L/R} = T(q_R^C)(f_{L/R} - p_R^C), \quad r_{vp} = T(q_R^C)\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}. \tag{14}$$

In case of coinciding body and camera frame origins and orientation $\Delta p_{cam} = 0$, $\Delta q_{cam} = 0$ and so the relations are:

$$r_{L/R} = T_{CB}T_{BR}(f_{L/R} - p_R), \quad r_{vp} = T_{CB}T_{BR}\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}. \tag{15}$$

In this case $T(q_R^C)$ simplifies to $T_{CB}T_{BR}$ where $T_{BR} = T_{RB}^T$ is the runway to body frame transformation and T_{CB} is the simplified rotation matrix from body to camera system representing only axis swapping as

$$\mathbf{T}_{CB} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}. \tag{16}$$

$f_L = \begin{bmatrix} 0 & -\frac{W}{2} & 0 \end{bmatrix}^T$ and $f_R = \begin{bmatrix} 0 & \frac{W}{2} & 0 \end{bmatrix}^T$ are the unknown left and right coordinates of threshold corners while $\begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T$ is the direction of the vanishing point in the runway frame (see Fig. 1). It is important to note that the runway threshold area is assumed to be flat resulting in the given positions of the corner points f_L and f_R. The perspective projection of these points onto the image plane is given by (17).

$$z_j = \frac{f}{r_{j,z}}\begin{bmatrix} r_{j,x} \\ r_{j,y} \end{bmatrix} + \nu_{z_j} = h_{z_j}(x,\nu) \quad j \in \{L, R, vp\} \tag{17}$$

Considering subscripts x, y, z as the coordinates of the vectors. This equation together with (14) shows how the uncertainty in the runway width estimation can affect the position and from that the velocity estimation. Surely, fluctuations in width estimation will cause an error in position estimation and slight changes in velocity estimates.

As mentioned before unknown runway characteristics require additional sensory information besides the camera and IMU. The supplementary sensor unit was chosen to be a barometric altitude sensor. The equation for the barometric sensor measurement can be expressed as follows:

$$z_{baro} = -p_z + p_{elevation} + \nu_{z_{baro}} = h_{baro}(x,\nu). \tag{18}$$

Here, $p_z = p_R^{(C)}(3)$ is the altitude of the aircraft in runway frame (negative in upward direction) and $p_{elevation}$ is the runway height above sea level (positive upward). The $p_{elevation}$ correction is required since the barometric sensor measures aircraft altitude above sea level.

Summarizing the system dynamics and measurements in the form of (6) results in:

$$\dot{x} = \begin{bmatrix} \dot{p}_R \\ \dot{v}_R \\ \dot{q} \\ \dot{b} \\ \dot{W} \end{bmatrix} = \underbrace{\begin{bmatrix} f_{p_R}(x,\eta) \\ f_{v_R}(x,\eta) \\ f_q(x,\eta) \\ f_b(x,\eta) \\ f_W(x,\eta) \end{bmatrix}}_{f(x,\eta)} + \underbrace{\begin{bmatrix} 0 \\ g_{v_R}(x) \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{g_1(x)} a_B + \underbrace{\begin{bmatrix} 0 \\ 0 \\ g_q(x) \\ 0 \\ 0 \end{bmatrix}}_{g_2(x)} \omega_B \tag{19}$$

$$z = \begin{bmatrix} z_L \\ z_R \\ z_{vp} \\ z_{baro} \end{bmatrix} = \underbrace{\begin{bmatrix} h_{z_L}(x,\nu) \\ h_{z_R}(x,\nu) \\ h_{z_{vp}}(x,\nu) \\ h_{baro}(x,\nu) \end{bmatrix}}_{h(x,\nu)}. \tag{20}$$

The targeted goal is to estimate the runway-relative position, velocity and orientation of the aircraft together with the width of the runway and the biases of the acceleration and angular rate sensors based-on these equations. However, considering any realistic vision sensor there is a significant delay which mostly comes from the image processing time. Since the processed image will include information about the past states it will corrupt the filter and result in inaccurate estimates (see later in Sections 5–7). Therefore the delay is compensated with delayed-ESKF algorithm from Watanabe et al. (2019) (briefly summarized in Section 4) by propagating the past measurement forward in time to make a correct measurement update.

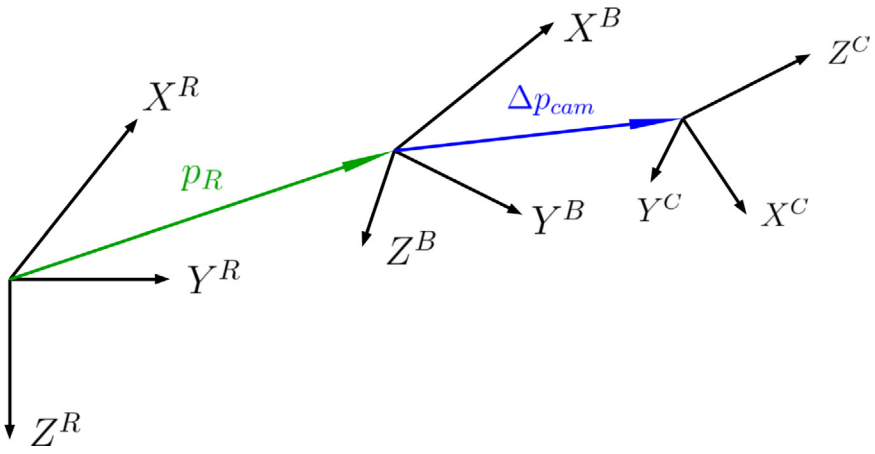

Fig. 4. Reference systems for the real data.

3. Observability of the nonlinear system

Before the design of any estimator the first task is to check system observability considering the dynamic Eqs. (19) and the selected measurements (20) of the system. As discussed before, the estimator cannot rely on GPS position or velocity information so there is no possibility for sensor selection. The observability of the system with monocular image and barometric altitude information (20) should be verified instead. As the delayed ESKF is derived in Watanabe et al. (2019) based-on Larsen et al. (1998) there is no need to consider delay effects in the evaluation of observability. This is because the time stamp of image measurements is known and so the effect of system dynamics between measurement acquisition and availability of the processed image can be considered directly in the measurement update of the filter. For the basic understanding of this process see Larsen et al. (1998) presenting the solution for the linear Kalman Filter.

Neglecting the random noises (η, ν) in the input affine form (6) the system can be formulated as:

$$\begin{aligned} \dot{x} &= f(x) + \sum_{i=1}^{m} g_i(x) u_i \\ z &= h(x). \end{aligned} \tag{21}$$

There is a broad range of literature about the observability of nonlinear systems. Montanari and Aguirre (2020) and Aguirre et al. (2018) give good overview about the possible methods. From these the Lie derivative-based observability calculation is the most straightforward and applicable calculating the observability co-distribution of the system. This method was extended to systems with inputs (such as ours represented by (21)) for example in Vidyasagar (1993). The algorithm can be summarized as in (22).

$$\begin{aligned} \psi_0 &= \mathcal{L}^0_{f,g_i} h(x) = h(x) \\ \psi_1 &= \mathcal{L}^1_{f,g_i} h(x) = \frac{\partial h}{\partial x}(x) f(x) + \sum_{i=1}^{m} \frac{\partial h}{\partial x}(x) g_i(x) u_i \\ \psi_2 &= \mathcal{L}^2_{f,g_i} h(x) = \frac{\partial \psi_1}{\partial x}(x,u) f(x) + \sum_{i=1}^{m} \frac{\partial \psi_1}{\partial x}(x,u) g_i(x) u_i \\ \psi &= \begin{bmatrix} \psi_0 \\ \psi_1 \\ \vdots \\ \psi_{N-1} \end{bmatrix} \quad dim(x) = N \\ \mathcal{O} &= \frac{\partial \psi}{\partial x} \\ rang(\mathcal{O}) &= N \end{aligned} \tag{22}$$

Here, $\mathcal{L}^j_{f,g_i}$ is the jth Lie derivative of a function with respect to the functions f, g_i. For the detailed formal definition see the equations in (22). The observability co-distribution $\mathcal{O}$ results by stacking the defined Lie derivatives $\psi_0 \dots \psi_{N-1}$ into a vector ψ and taking its spatial derivative with respect to the state variable x. If the rank of $\mathcal{O}$ equals the dimension of the state space N at every (x, u) point then the system is globally observable.

If $\mathcal{O}$ is symbolically calculated and is full column rank (in case of multiple measurements there can be more rows than columns) then the system can be globally observable. In case if the symbolic matrix is rank deficient then the system is globally unobservable. Note that here symbolic calculation means the symbolic substitution of functions $h(x), f(x), g_i(x)$ (from (19) and (20)) and is not related to the symbolic observability methods considering only the structure of these functions (Bianco-Martinez et al., 2015; Letellier & Aguirre, 2009).

The symbolic calculation is done by applying the Matlab Symbolic Toolbox substituting Eqs. (7) to (11) and (17) to (18) neglecting the effect of camera orientation (Δq_{cam}) and position (Δp_{cam}) relative to the body system as these are known constant parameters so should not affect observability. Symbolically calculating the observability co-distribution according to (22) with $z_{vp}, z_{L/R}, z_{baro}$ measurements ($\mathcal{O}_1$) gives full rank (rank 17) so there is no global unobservability problem. Note that there are 17 states summing up position (3), velocity (3), quaternion (4), acceleration bias (3), angular rate bias (3) and runway width (1) dimensions. The rank is the same with $z_{L/R}, z_{baro}$ measurements ($\mathcal{O}_2$) so the measurement of the vanishing point is not mandatory. However, with $z_{vp}, z_{L/R}$ measurement only ($\mathcal{O}_3$) the symbolic rank is 16 meaning global unobservability so the barometric measurement is mandatory.

However, the possible local loss of observability should also be examined theoretically for all (x, u) pair and parameter values as the nonlinear relations can be singular or vanish at some points. Practically only the critical points should be examined but there is no systematic way to generate them thus the chance of false evaluation is large. So formally examining the singularities and vanishing values can be faster and safer.

From the literature the graph-based approach (Aguirre et al., 2018; Letellier et al., 2018) can help in the detection of dependencies and in detailed examination of the singularities and vanishing connections. So at first, the inference diagram of the system was drawn (see Fig. 5) considering the Jacobian of $h(x)$ representing the effect of state changes to the output, the Jacobian of $f(x)$ representing the effect of state changes to the states and $g_i(x)$ as the effect of inputs to the states (see Aguirre et al. (2018)). In the diagram in Fig. 5 non-singular, never vanishing connections are denoted with solid lines while nonlinear, possibly singular and/or vanishing ones with dashed lines. I_3 is the three dimensional unit matrix, e_3 is its last row, z_{ba} is the measured barometric altitude $B_a(q, b_a)$ and $B_\omega(b_\omega)$ are matrices including the quaternion and sensor bias values. Both of them can be zero in the ideal case with zero biases.

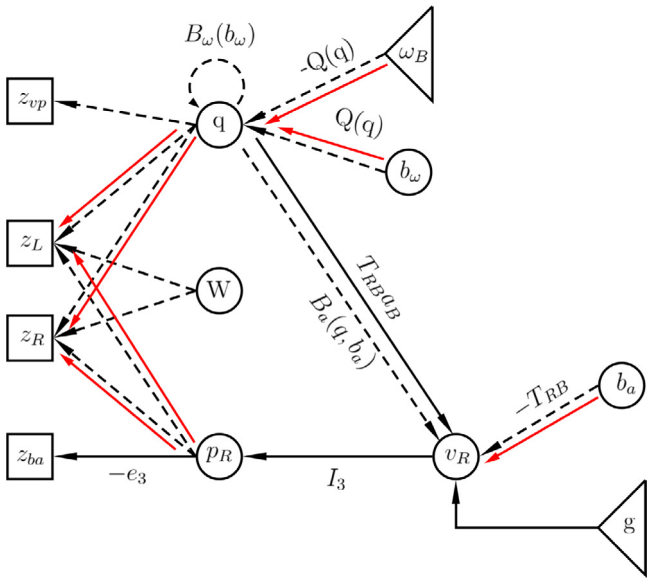

Fig. 5. Inference diagram of the system with image and barometric altitude measurements. Black lines (solid: never vanishing, dashed: possibly singular and vanishing) are the initial connections, solid red lines are the detected non-singular and never vanishing ones.

$g_i(x)$ give the connections with ω_B and a_B through $-Q(q)$ (9) and T_{RB}(8). T_{RB} is an orthogonal rotation matrix which is never singular and the acceleration measurement a_B includes gravity and so it is never zero. That is why there is always a connection between the orientation q and the velocity v_R (solid line). As $-Q(q)$ gives the derivative of the quaternion from the angular rate this connection is never zero as any nonzero angular rate should cause a change of the orientation. This is also true for the $Q(q)b_\omega$ part its only zero when $b_\omega = 0$.

The third coordinate of the position is directly measured through the barometric altitude z_{ba} so this connection never vanishes. However, the effects of p_R, q and W to $z_{L/R}$ and z_{vp} image measurements should be examined in detail.

As W is the constant runway width its time- and state-dependence is zero and hence its effect should be evaluated through $h(x)$ instead of the Jacobian of $h(x)$. The singularity of $h_{z_j}(x)$, $j \in \{L, R\}$ from (17) occurs if $r_{j,z} = 0$. This is the case when the feature point is in the $X^C - Y^C$ camera plane (see Fig. 1) which case the feature point is outside camera field of view (which is 67° for the wide angle camera in the VISION project). Consequently the runway width effect is never singular. The next step is to evaluate if this connection can vanish. Examining (17) by substituting the terms from (14) shows that the runway width effect can vanish either on z_L when the camera points directly to f_L or on z_R when the camera points directly to f_R but never together. This means that there will always be runway width effect on the measured corner point images but sometimes only on one of them.

After examining the runway width effect the effects from p_R position and q quaternion should be examined through the Jacobian of $h_{z_j}(x)$:

$$\frac{\partial h_{z_j}}{\partial x_i}(x) = \begin{bmatrix} -\frac{f r_{j,x}}{r_{j,z}^2}\frac{\partial r_{j,z}}{\partial x_i} + \frac{f}{r_{j,z}}\frac{\partial r_{j,x}}{\partial x_i} \\ -\frac{f r_{j,y}}{r_{j,z}^2}\frac{\partial r_{j,z}}{\partial x_i} + \frac{f}{r_{j,z}}\frac{\partial r_{j,y}}{\partial x_i} \end{bmatrix} \quad j \in \{L, R, vp\}, x_i \in \{p_R, q\}. \tag{23}$$

Here f is the focal length of the camera. These expressions again will be singular when $r_{j,z} = 0$ which is not possible in realistic situations (limited camera field of view). To detect if these connections can vanish vanishing of $r_{j,x}, r_{j,y}, \frac{\partial r_{j,z}}{\partial x_i}, \frac{\partial r_{j,x}}{\partial x_i}, \frac{\partial r_{j,y}}{\partial x_i}$ should be examined. The detailed examination is rather lengthy so it is omitted here due to space constraints. The final conclusions are listed as:

- The position change effect on $z_{L/R}$ will never vanish.
- The quaternion change effect on $z_{L/R}$ will never vanish.
- The quaternion change effect on z_{vp} can vanish in special cases:
  1. Z^C camera axis points to left runway corner point (f_L) and aircraft motion in $Z^C - Y^C$ camera plane.
  2. Z^C camera axis points to left runway corner point (f_L) and aircraft motion in $Z^C - X^C$ camera plane.
  3. $q_{s1} = \begin{bmatrix}1 & 0 & 0 & 0\end{bmatrix}$ special quaternion orientation and free other coordinates.

Of course the Z^C camera axis can also point towards the right runway corner point (f_R) instead of the left, but this does not change the observability results. Realistic numerical values of the states for the above $1-3$ cases are generated as follows:

$$1: \; p_R = \begin{bmatrix}-200\\-15\\-10.4816\end{bmatrix}, \; v_R = \begin{bmatrix}30\\0\\0\end{bmatrix}, \; q = \begin{bmatrix}0.9997\\0\\-0.0262\\0\end{bmatrix}, \; b_a = 0, \; b_\omega = 0$$

$$W = 30, \; a_B = \begin{bmatrix}0.5134\\0\\9.7966\end{bmatrix}, \; \omega_B = 0$$

$$2: \; p_R = \begin{bmatrix}-200\\-15\\-10.4816\end{bmatrix}, \; v_R = \begin{bmatrix}27.9616\\2\\1.4654\end{bmatrix}, \; q = \begin{bmatrix}0.9997\\0\\-0.0262\\0\end{bmatrix}, \; b_a = 0, \; b_\omega = 0$$

$$W = 30, \; a_B = \begin{bmatrix}0.5134\\0\\9.7966\end{bmatrix}, \; \omega_B = 0$$

$$3: \; p_R = \begin{bmatrix}-200\\-15\\-10.4816\end{bmatrix}, \; v_R = \begin{bmatrix}30\\0\\0\end{bmatrix}, \; q = \begin{bmatrix}1\\0\\0\\0\end{bmatrix}, \; b_a = 0, \; b_\omega = 0$$

$$W = 30, \; a_B = \begin{bmatrix}0\\0\\9.81\end{bmatrix}, \; \omega_B = 0.$$

Checking numerically $\mathcal{O}_1$ and $\mathcal{O}_2$ with these values also gives full rank so the system is globally observable from $z_{vp}, z_{L/R}, z_{baro}$ or from $z_{L/R}, z_{baro}$. To have more redundancy in measurements $z_{vp}, z_{L/R}, z_{baro}$ are applied in the development of the delayed-ESKF.

4. Brief summary of the delayed-ESKF algorithm

For the targeted estimation task the delayed measurement ESKF algorithm developed in Watanabe et al. (2019) is applied. It is briefly overviewed in this section.

4.1. ESKF process

In the ESKF framework the nonlinear system in consideration (6) will be decomposed into a nominal system (subscript n) and an error system (denoted with δ) such that the nominal system does not include any uncertainty. First, the nominal state system can be defined as

$$\begin{cases} \dot{x}_n = f(x_n, 0) + \sum_{i=1}^{m} g_i(x_n)u_i \\ z_n = h(x_n, 0). \end{cases} \tag{24}$$

The error state is defined as $x = x_n \oplus \delta x$ with an injection operator $\oplus$. Here, the same state decomposition is applied as in Watanabe et al. (2019) with a linear decomposition $W = W_n + \delta W$ for the additional runway width state:

$$x = \begin{bmatrix} p_R \\ v_R \\ q \\ b_a \\ b_\omega \\ W \end{bmatrix} = \begin{bmatrix} p_{R_n} + T_{RB}\delta p_R \\ v_{R_n} + T_{RB}\delta v_R \\ q_n \otimes q(\delta\theta) \\ b_{a_n} + \delta b_a \\ b_{\omega_n} + \delta b_\omega \\ W_n + \delta W \end{bmatrix} = \begin{bmatrix} p_{R_n} \\ v_{R_n} \\ q_n \\ b_{a_n} \\ b_{\omega_n} \\ W_n \end{bmatrix} \oplus \begin{bmatrix} \delta p_R \\ \delta v_R \\ \delta\theta \\ \delta b_a \\ \delta b_\omega \\ \delta W \end{bmatrix} = x_n \oplus \delta x. \tag{25}$$

Here, $q(\theta)$ represents a quaternion defined from a rotation vector $\theta \in \mathbb{R}^3$ (See Sola (2017)). It should be noted that because of this approximation of the rotation δx has one-dimension less ($dim(\delta x) = 16$) than the original and nominal state vectors ($dim(x) = dim(x_n) = 17$).

Finally, the error system becomes the following linear one

$$\begin{cases} \delta\dot{x} = A(x_n, u)\delta x + B(x_n)\eta \\ \delta z = z - z_n = C(x_n)\delta x + D(x_n)\nu. \end{cases} \tag{26}$$

Here, the matrix A is given with $a_n = a_B - b_{a_n}$ and $\omega_n = \omega_B - b_{\omega_n}$ and the matrix B becomes constant as follows

$$A(x_n, u) = \begin{bmatrix} -[\omega_n\times] & I_{3\times3} & O_{3\times3} & O_{3\times3} & O_{3\times3} & O_{3\times1} \\ O_{3\times3} & -[\omega_n\times] & -[a_n\times] & I_{3\times3} & O_{3\times3} & O_{3\times1} \\ O_{3\times3} & O_{3\times3} & -[\omega_n\times] & -I_{3\times3} & O_{3\times3} & O_{3\times1} \\ O_{7\times3} & O_{7\times3} & O_{7\times3} & O_{7\times3} & O_{7\times3} & O_{7\times1} \end{bmatrix},$$

$$B(x_n) = \begin{bmatrix} O_{3\times13} \\ I_{13\times13} \end{bmatrix}.$$

Here, $[\upsilon\times]$ means the matrix representation of vector cross product with any υ vector. The error system measurement δz for the image coordinates of left and right corner positions is given by

$$\delta z_{L/R} = z_{L/R} - h_{z_{L/R}}(x_n, 0) = H(r_{(L/R)_n})\delta r_{L/R} + \nu_{L/R}. \tag{27}$$

The Jacobian matrix $H(r_{(L/R)})$ of the image projection function results as:

$$H(r_{(L/R)}) = \frac{f}{r_{(L/R)z}}\left[I_{2\times2} - \frac{1}{r_{(L/R)z}}\begin{bmatrix} r_{(L/R)x} \\ r_{(L/R)y} \end{bmatrix}\right].$$

From (14) and (15):

$$r_{L/R_n} = T(q^C_{R,n})(f_{L/R_n} - p^C_{R_n}) = T_{CB}\left(T_{BR,n}\left(f_{L/R_n} - p_{R_n}\right) - \Delta p_{cam}\right).$$

Then its difference from the real position $r_{L/R}$ is affine approximated by

$$\delta r_{L/R} = r_{L/R} - r_{L/R_n} =$$
$$T_{CB}\left(-\delta p_R + \left[T_{BR,n}\left(f_{L/R_n} - p_{R_n}\right)\times\right]\delta\theta \mp T_{BR,n}\begin{bmatrix} 0 \\ \delta W/2 \\ 0 \end{bmatrix}\right). \tag{28}$$

In a similar manner, the error system measurement for the vanishing point can be obtained as follows:

$$\delta z_{vp} = z_{vp} - h_{z_{vp}}(x_n, 0) = H(r_{vp_n})\delta r_{vp} + \nu_{vp}, \quad r_{vp_n} = T_{CB}T_{BR,n}e_1$$
$$\delta r_{vp} = r_{vp} - r_{vp_n} = T_{CB}\left[T_{BR,n}e_1\times\right]\delta\theta.$$

Here, $e_1 = \begin{bmatrix}1 & 0 & 0\end{bmatrix}^T$. The error system measurement for the barometric altitude is given by

$$\delta z_{baro} = z_{baro} - h_{baro}(x_n, 0) = -e_3^T T_{RB,n}\delta p_R + \nu_{baro}.$$

Here, $e_3 = \begin{bmatrix}0 & 0 & 1\end{bmatrix}^T$. Hence the matrix C in (26) becomes (with $e_2 = \begin{bmatrix}0 & 1 & 0\end{bmatrix}^T$)

$$C(x_n) = \begin{bmatrix} H(r_{L_n})T_{CB}\left[-I_{3\times3} \quad O_{3\times3} \quad \left[T_{BR,n}\left(f_{L_n} - p_{R_n}\right)\times\right] \quad O_{6\times3} \quad -\frac{1}{2}T_{BR,n}e_2\right] \\ H(r_{R_n})T_{CB}\left[-I_{3\times3} \quad O_{3\times3} \quad \left[T_{BR,n}\left(f_{R_n} - p_{R_n}\right)\times\right] \quad O_{6\times3} \quad \frac{1}{2}T_{BR,n}e_2\right] \\ H(r_{vp_n})T_{CB}\left[O_{3\times3} \quad O_{3\times3} \quad \left[T_{BR,n}e_1\times\right] \quad O_{6\times3} \quad O_{1\times3}\right] \\ \left[-e_3^T T_{RB,n} \quad O_{1\times13}\right] \end{bmatrix}.$$

Finally, $D(x_n) = I_{7\times7}$.

Since the nominal state is propagated in a deterministic manner the idea of ESKF is to apply a linear KF to estimate the error state δx instead of the true state x. In the ESKF process after each KF correction the estimated error state $\delta\hat{x}$ is injected into the nominal state and then it is reset to zero. The estimation error covariance δP is also reset accordingly. This injection and reset operation makes the error state to operate always around the origin and so its linear approximation holds.

The local observability of this linear error system can be checked with the rank of the observability matrix defined with $A(x_n, u)$ and $C(x_n)$. As these are pointwise linearizations of the nonlinear dynamics for which global observability was proven in Section 3 there will be no observability problem.

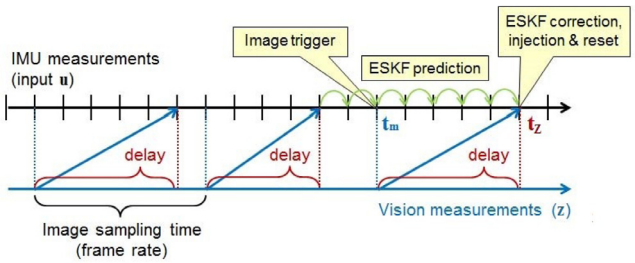

Fig. 6. ESKF process timeline with delayed measurement.

4.2. Delayed ESKF process

In this filter it is considered that the measurement z is taken at a time instant t_m and arrives at $t_z > t_m$ with delay. As stated earlier this is the case for the vision sensor measurements due to image processing time which can vary for each image. But a nice thing about the vision sensor is that the time of image acquisition t_m can be controlled by an image trigger thus it is known. The delayed-ESKF algorithm is designed based-on this fact. For simplicity this paper assumes that the time delay does not exceed the image sampling time. Fig. 6 shows the ESKF process timeline with the delayed vision measurement.

Let $x_n(t_m)$ be the nominal state obtained at time t_m. Let $\delta\hat{x}^-(t) = 0$ and $\delta P^-(t)$ be the predicted error state estimate and its estimation error covariance at a time t ($t_m < t \leq t_z$). It should be noted that the predicted error state always remains zero. The measurement which arrives at t_z contains an information on the state at t_m so the mathematical model of the delayed measurement is:

$$y(t_z) = z_n(t_z) + \delta z(t_z) = h(x_n(t_m), 0) + C(x_n(t_m))\delta x(t_m) + D(x_n(t_m))\nu(t_z).$$

A modified ESKF correction process can be applied to directly update the predicted state at t_z from this delayed measurement according to Larsen et al. (1998) considering the effect of system dynamics forward in time between t_m and t_z:

$$\begin{aligned} \delta\hat{x}(t_z) &= \delta\hat{x}^-(t_z) + K(t_z)(z(t_z) - z_n(t_z)) = K(t_z)(z(t_z) - h(x_n(t_m), 0)) \\ \delta P(t_z) &= \delta P^-(t_z) - K(t_z)C(x_n(t_m))\delta P^{-T}_{zm} \\ K(t_z) &= \delta P^-_{zm}C(x_n(t_m))\cdot \\ &\quad \left(C(x_n(t_m))\delta P^-(t_m)C^T(x_n(t_m)) + D(x_n(t_m))R_\nu(t_z)D^T(x_n(t_m))\right)^{-1}. \end{aligned} \tag{29}$$

Here, $\delta P^-_{zm} = \mathbb{E}\left[\delta\hat{x}^-(t_z)\delta\hat{x}^{-T}(t_m)\right]$ is the correlation between the predicted error state at t_z and t_m and $R_\nu(t_z) = \mathbb{E}\left[\nu(t_z)\nu^T(t_z)\right]$ is the measurement noise covariance. In the presented case as there is no intermittent other measurement the correlation matrix simply becomes $\delta P^-_{zm} = \Phi_{(t_m,t_z)}\delta P^-(t_m)$ with a state transition matrix $\Phi_{(t_m,t_z)}$ of (26) from t_m to t_z. This is a function of the nominal state x_n and the IMU input u over the time interval from t_m to t_z.

Knowledge of the image trigger time allows the estimation filter to avoid storage of nominal state and IMU input histories and so the computational load will be distributed over the time steps. When the image trigger signal is received at t_m one can calculate and store $z_n(t_z) = h(x_n(t_m), 0)$, $C(x_n(t_m))$ and $D(x_n(t_m))$, and initialize the state transition matrix by $\Phi_{(t_m,t_m)} = I$. Then upon each KF prediction when a new IMU measurement is received the state transition matrix is updated by $\Phi_{(t_m,t+\Delta t)} = e^{A(x_n(t),u(t))\Delta t}\Phi_{(t_m,t)}$. When the vision measurement $z(t_z)$ arrives at t_z and $R_\nu(t_z)$ is determined (in case if time-dependent) all the necessary matrices to calculate the Kalman gain $K(t_z)$ and execute (29) are available.

When considering the time-delay of the image-based measurements the first nine rows of the measurement matrix $C(x_n)$ in (26) will be modified multiplying form the right with the inverse of the state-transition matrix $\Phi_{(t_m,t)}$. Since $\Phi_{(t_m,t)}$ is full rank this will not affect local observability of the system near the aircraft approach trajectory.

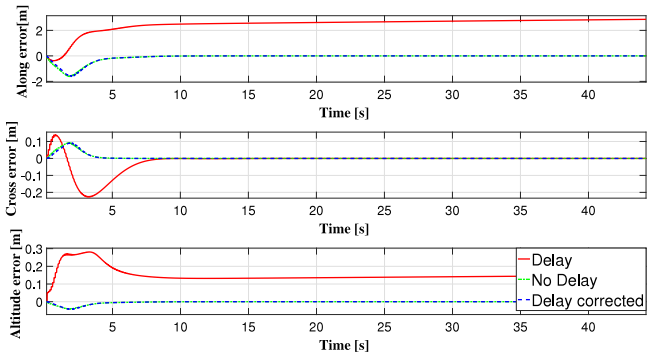

Fig. 7. Deviation of estimated position from simulated reference values in case of ideal data.

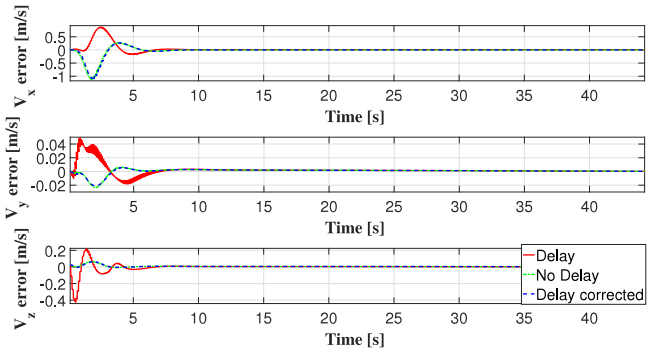

Fig. 8. Deviation of estimated velocity from simulated reference values in case of ideal data.

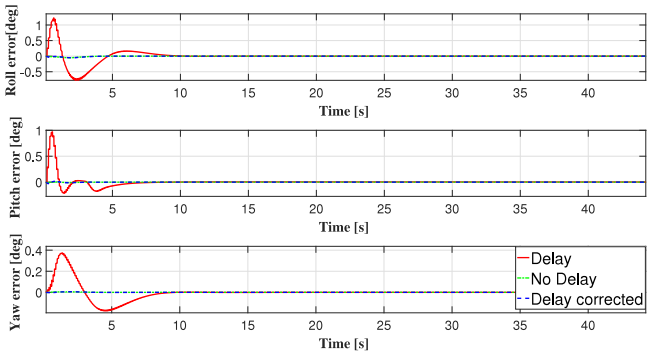

Fig. 9. Deviation of estimated orientation from simulated reference values in case of ideal data.

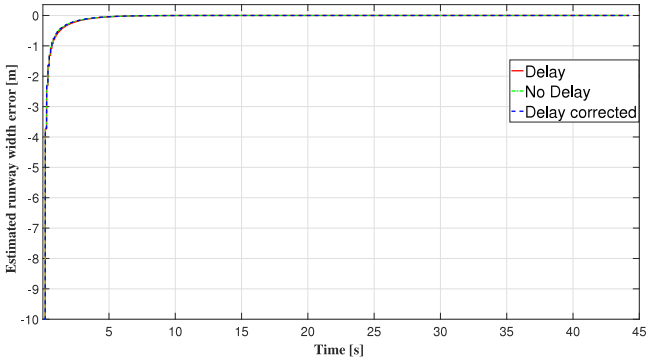

Fig. 10. Deviation of estimated runway width from real value in case of ideal data.

5. Test on ideal simulated data without and with delay compensation

After formulating the system equations in Section 2, checking observability in Section 3 and designing the delayed-ESKF algorithm in Section 4 the developed estimator is tuned and applied to different data sets to gradually test its capabilities.

First, it is tested with Matlab/Simulink simulation-generated sensor and flight data. The Simulink simulation includes the model of the K-50 test aircraft (see Fig. 3) and an ILS model from the VISION project. The ILS is applied to guide the aircraft towards the runway without closing the control loop with the vision-based estimator. The simulation contains a block that generates the aircraft runway-relative position, velocity and orientation. These are considered as reference data in the evaluation of the estimator.

As it was stated earlier in Section 4 the IMU, the barometric sensor and the image processing unit operate with different measurement frequency. In the implemented ESKF algorithm the correction step is only applied when there is a new processed image. Otherwise only the prediction steps are propagated. In the simulation the IMU unit frequency was 100 Hz while the camera frequency was set to 10 Hz and the barometric measurement was simply modeled by the runway-relative altitude. It was assumed to have the same frequency and the same amount of delay as the camera sensor because no measurement update can be done from solely the barometric measurement due to loss of observability.

First, the estimator was tested with ideal data sets without sensor bias and noise. The goal was to test the ESKF algorithm with non-delayed and delayed image data and the delayed-ESKF algorithm with delayed image data to see the effect of delay compensation and check if the filter implementations are flawless. In the simulation the initial position had both vertical and horizontal offset from the glide slope and localizer causing transients in aircraft motion and so a dynamic situation for the position estimation. Figs. 7–10 show the errors of the estimated states. The initial estimated states were set as perfectly accurate except for the runway width which was set to 20.22 m with 10 m error from the real value (30.22 m).

In Figs. 7–10 the *No Delay* legend refers to the case when no image processing delay is considered and the ESKF is applied. The *Delay* case shows how the applied image processing delay affects the estimates of the ESKF filter. Finally, in the *Delay corrected* case the image processing delay is applied to the data and handled by the delayed-ESKF algorithm.

The simulated delay was set to 0.09 s considering 10 Hz image frequency in order to correctly model the real scenario. In the *Delay* case the estimation lagged behind the real values. The biggest difference was experienced for the position estimates. The deviation scales with the length of the camera delay. For example in case of the Along error a 0.09 s delay time results in approximately 3 m position estimation error (see Fig. 7) which is consistent with the 30 m/s along speed and the almost 0.1 s delay. The delayed camera measurements also affect the early transient period in both length and magnitude as it can be seen in the figures. Note that the position values are the most sensitive to the delay as their error is directly proportional to the velocity.

Therefore, the *Along* and *Altitude* values are the most susceptible to the delay time for the aircraft landing scenario. The other states such as orientation, velocity and runway width are almost constant during the simulation so their estimation errors are the same for both estimators apart from the early transient period. Applying the delayed-ESKF algorithm it can be seen that the algorithm corrects the delay effects and provides almost inseparably the same results as the ESKF with non-delayed data. In the *No delay* and *Delay corrected* cases after the early transient period ends the persisting error for all variables is approximately zero. Therefore, it can be concluded that the delayed-ESKF algorithm is capable of estimating all the desired states of the aircraft and the algorithm implementations are flawless.

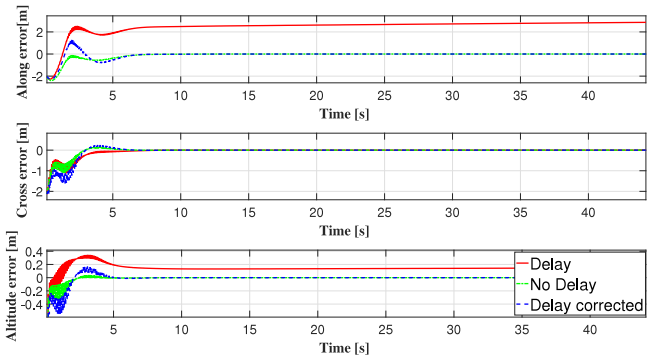

Fig. 11. Deviation of estimated position from simulated reference values in case of realistic data.

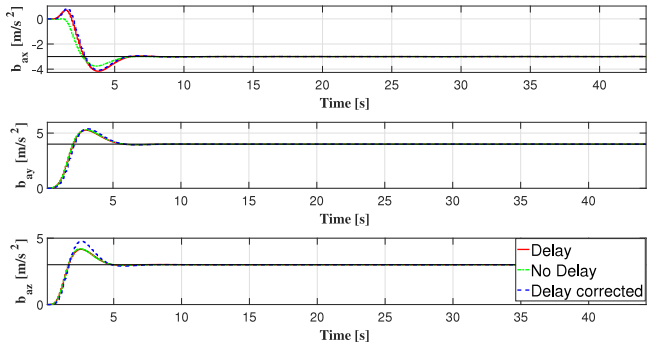

Fig. 12. Acceleration bias estimation results in case of realistic data.

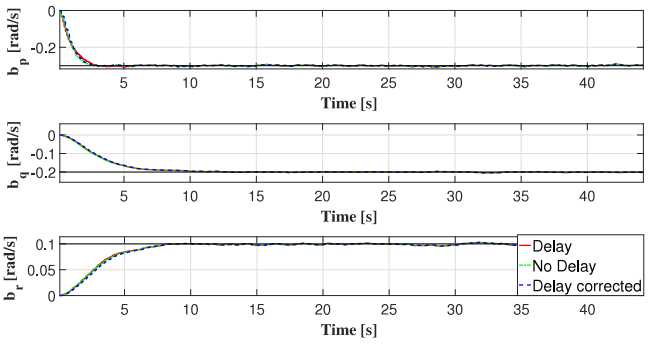

Fig. 13. Angular rate bias estimation results in case of realistic data.

Table 2
'Steady state' Root mean square error of the estimated states for Non-delayed, Delayed and Delay corrected cases.

Variable	No delay	Delay	Corrected
$V_x[\frac{m}{s}]$	0.003	0.01	0.004
$V_y[\frac{m}{s}]$	0.02	0.021	0.036
$V_z[\frac{m}{s}]$	0.021	0.021	0.035
$Along[m]$	0.003	2.7	0.003
$Cross[m]$	0.001	0.001	0.001
$Alt[m]$	0.001	0.14	0.001
$Roll[deg]$	0.037	0.037	0.07
$Pitch[deg]$	0.038	0.038	0.066
$Yaw[deg]$	0.04	0.04	0.07
$Accbias[\frac{m}{s^2}]$	0.002	0.003	0.002
$Angbias[\frac{rad}{s}]$	0.002	0.002	0.002
$Runway[m]$	0.001	0.001	0.001

6. Test on realistic simulated data without and with delay compensation

After validating the ESKF and delayed-ESKF algorithms with ideal simulated data several realistic data sets were generated including sensor noises and biases to further verify proper operation.

The primary goals were to test the noise tolerance and the precision of bias estimation. The latter cannot be verified on real flight data as the sensor biases are unknown. Furthermore, offset values were added to the initial estimated states to evaluate filter convergence. The offset values for the position, velocity, orientation and runway width were 2 m, 1 m/s, 1° and −10 m respectively. They were applied for every component of their vectors. The simulated sensor bias values were $b_a = \begin{bmatrix}-3 & 4 & 3\end{bmatrix}$ m/s^2 for the accelerometer and $b_\omega = \begin{bmatrix}-0.3 & -0.2 & 0.1\end{bmatrix}$ rad/s for the gyroscope.

The added noise was zero mean, gaussian with a variance of 0.001 (m/s)^2 for accelerations and 0.001 (rad/s)^2 for angular rates. Again, ESKF and delayed-ESKF were both tested considering non-delayed and delayed image data. The legends *Delay*, *No Delay* and *Delay corrected* in Figs. 11–13 follow the same terminology as in Section 5. Fig. 11 shows that the position estimation errors are similar to the ideal data case.

Verifying correct sensor bias estimation is an important step and can be observed in Figs. 12 and 13. They show that the algorithm correctly estimates the simulated sensor biases both for the accelerometer and gyroscope.

Table 2 shows the root mean square errors (RMSEs) of the estimated states after the transient period. The results indicate the same conclusions as in Section 5 the *Along* and *Altitude* values are the most sensitive to the delay. The other states have nearly zero RMSE values in all cases. It is important to note that longer image processing times i.e. larger delay would result in greater steady state errors without delay compensation.

These estimation results show that an automated landing scenario is executable since the average position error is about 0.03 m for the cross track and altitude while the speed of the aircraft is estimated with a precision of about 0.02 m/s. This applicability question will be examined in detail also for real flight data. The simulation tests show that the developed algorithm runs with adequate precision. Thus accurate information can be provided for the autopilot system in case of an automated landing scenario or for fault detection. The next step is estimator evaluation with real flight data summarized in the following Section.

7. Test on real flight data with delay compensation

In this section the ESKF and delayed-ESKF filters are evaluated on real flight data with unknown runway width. Flight data sets were collected in frame of the VISION EU H2020 research project at a runway in Septfonds, France (see Fig. 2) during the summer of 2019. The K-50 aircraft was equipped with all the necessary sensors (IMU, GPS SBAS, Barometric, Camera). The filters are compared to GPS with Satellite Based Augmentation System (SBAS) measurements.

Guilhamu (2019) gave a detailed presentation about AIRBUS Fleet Readiness for GBAS/SBAS systems. The presentation declares that the SBAS and Ground Based Augmentation System (GBAS) technology is ready to be applied in automated landing as it meets the precision requirements. Therefore, the SBAS technology can be considered accurate enough for autopilot systems. Tessier et al. (2017) also declares that GNSS with augmentation systems such as SBAS could be considered as the successor of the ILS regarding precision levels. These references justify our choice of using the GPS SBAS measurements as reference values to evaluate the precision of the vision-based algorithms.

Algorithm tuning was done by trial and error starting from realistic noise covariances partially based-on real measurements. The final covariances are presented and explained in Appendix.

The objective of tuning was to provide similarly accurate results in position and velocity estimation as the GPS SBAS system. In the

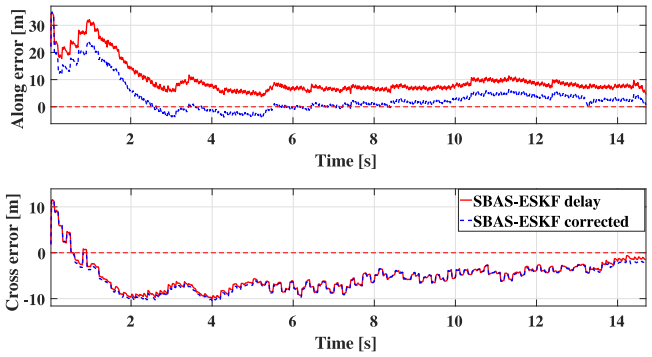

Fig. 14. Along and Cross deviation of estimates from SBAS reference values in case of real flight data.

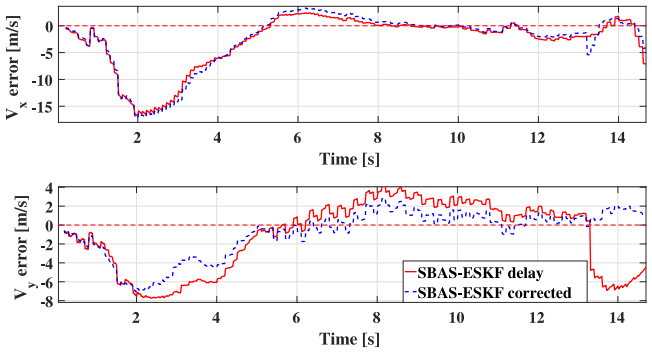

Fig. 15. V_x and V_y estimates deviation from SBAS reference values in case of real flight data.

test flights the IMU and the GPS SBAS unit ran at 50 Hz and 20 Hz respectively. The camera was slower with close to 10 Hz frequency. Because of the different sampling frequencies an interpolation was applied to the SBAS values to match 50 Hz. The barometric sensor had a frequency of approximately 2 Hz. Therefore extrapolation was performed from the last altitude based on system dynamics in Eqs. (7) and (8) so that the barometer operates at the same frequency as the camera.

Again both algorithms were executed but now only for delayed data as there is no delay-free data set in the real flight tests. The initial values of the estimated states were set based on GPS SBAS data assuming a scenario where GPS SBAS signal is available at the beginning of landing. Later the SBAS correction signal can be lost and in that case the IMU–Camera–Barometric estimator should take over. The information about the runway was assumed to be unknown setting the initial estimate of its width to 20.3 m instead of the actual 25.3 m. It is worth noting that the knowledge of precise GPS SBAS position can make this initialization more accurate by obtaining a more realistic initial guess despite the unknown geo-location of the threshold line.

Figs. 14–17 display the results with one of the test flight data sets. The estimates are compared to the reference GPS SBAS values so SBAS is included in the legends of the estimation errors. The legend *SBAS-ESKF delay* refers to the results when the image processing time delay was not considered in the ESKF and therefore it corrupts the estimation. On the other hand *SBAS-ESKF corrected* displays how the delayed-ESKF algorithm corrects the estimates. Fig. 17 shows also the pairwise differences of the measured altitudes labeled as *SBAS-baro* and *SBAS-Camera*. Those signals correspond to the deviation between the SBAS values and the barometric or pure visual measurements obtained according to Hiba et al. (2021).

The relatively large inaccuracy in the initial runway width estimate resulted in large errors observed at the beginning especially in position

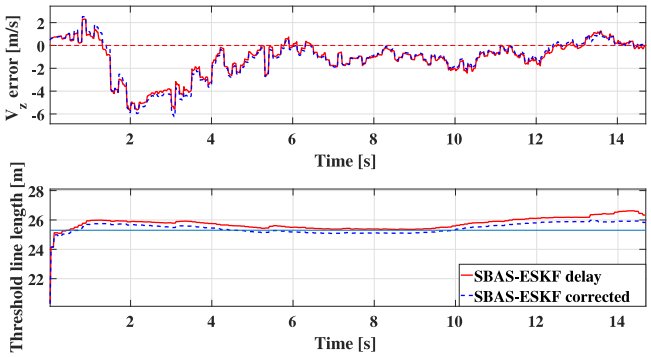

Fig. 16. Estimated V_z deviation from SBAS reference values and threshold line length (runway width) estimate in case of real flight data.

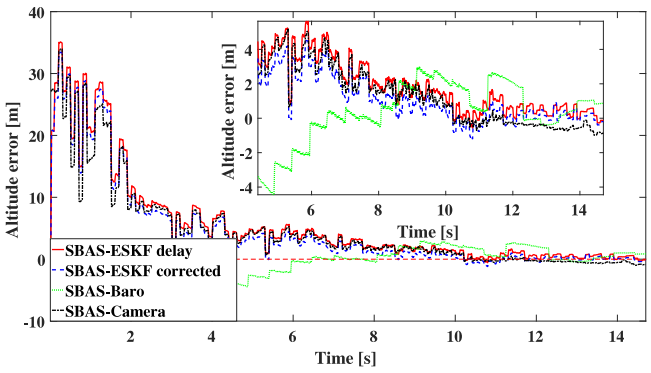

Fig. 17. Altitude deviation from SBAS reference values in case of real flight data.

Table 3
'Steady state' RMSE values for Delayed and Delay corrected cases with real flight data.

Variable	Delay	Corrected
$V_x[\frac{m}{s}]$	1.567	1.40
$V_y[\frac{m}{s}]$	3.38	1.43
$V_z[\frac{m}{s}]$	1.63	1.51
$Along[m]$	8.03	3.25
$Cross[m]$	4.38	4.47
$Alt[m]$	1.32	0.80
$Runway[m]$	0.71	0.41

estimation. Fig. 14 shows that at the beginning even about 25 m errors can occur in the Along position. Since the runway width determines the image scaling its estimation error directly influences the position estimation through Eq. (17). The slow convergence of the sensor bias values along with the early uncertainty of the runway width estimation causes the filters to provide unreliable early estimates.

Similarly to the results presented in Sections 5 and 6 the states most affected by the image delay are the *Along* and *Altitude* positions. Again, the velocity state is less susceptible to the delay issues as it is almost constant. During the test flights the delay time was approximately 0.107 s which mainly originates from the image processing time and minor additional delays such as UART (Universal Asynchronous Receiver–Transmitter) delay, image grab delay and software trigger delay. Fig. 17 shows that the altitude estimates are closer to the image data than to the barometric measurement as the estimator was weighted to rely mainly on the image data. As the uncertainty in image measurements decreases by approaching the runway the estimation errors also decrease.

Table 3 shows the RMSE values of the estimations compared to the GPS SBAS reference. The steady state errors for the V_z and *Altitude* states are approximately 1.5 m/s and 0.8 m respectively which can be acceptable in a landing scenario.

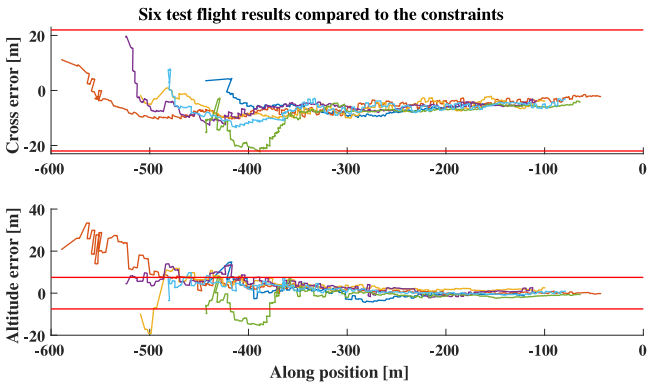

Fig. 18. Cross and Altitude error compared to the control constraints (continuous horizontal red lines).

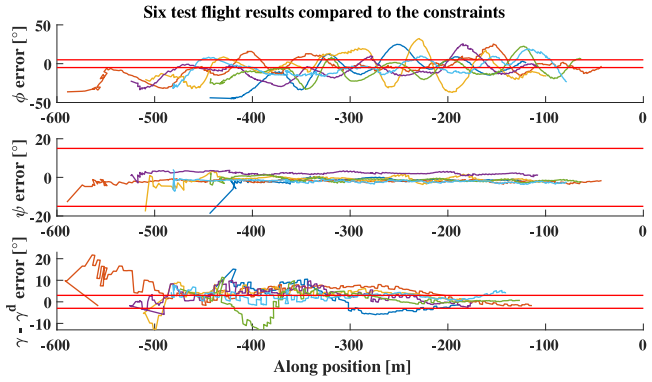

Fig. 19. Runway-relative roll (ϕ), yaw (ψ) and flight path angle error compared to the control constraints (continuous horizontal red lines).

Burlion and Kolmanovsky (2020) proposes a vision-based constrained control solution for final approach scenarios. That article provides a set of control constraints proposed by Airbus in the framework of VISIOLAND project. The mentioned constraints can be written in the following form:

$$-22 \text{ m} \leq \Delta p_y \leq 22 \text{ m} \tag{30}$$

$$-7.5 \text{ m} \leq \Delta p_z \leq 7.5 \text{ m} \tag{31}$$

$$-3 \text{ deg} \leq \gamma - \gamma^d \leq 3 \text{ deg} \tag{32}$$

$$-15 \text{ deg} \leq \psi \leq 15 \text{ deg} \tag{33}$$

$$-5 \text{ deg} \leq \phi \leq 5 \text{ deg}. \tag{34}$$

Where Δp_y and Δp_z refer to the cross and altitude errors. ψ and ϕ are the yaw and roll angle errors. The deviation from the designated γ^d flight path angle is also evaluated. It is considered to be the GPS SBAS-based actual flight path angle.

In order to complete the approach phase the aircraft state deviations should be inside these limits. Essentially if the estimates provided by the delayed-ESKF algorithm fall into the mentioned intervals relative to the GPS SBAS measurements the auto landing can be executed by closing the control loop with its results. Figs. 18 and 19 display the estimation errors from the reference GPS SBAS values for all six test flights along with the limit sets given in Eqs. (30)–(34). Note that legends listing the flight test cases are not provided as the main message of the figures is that the errors are inside the limits most of the time in all flights. Fig. 18 shows that the cross and altitude errors fall into the designated intervals for every test flight after a short transient period and are well inside the limits after convergence. Due to the fact that the GPS SBAS is unable to provide information about the aircraft orientation the reference attitude values were retrieved from an IMU-based estimator algorithm of the flight avionics system. Unfortunately the roll angle has large uncertainty giving sometimes unrealistic values as it is estimated from the acceleration vector. Consequently the roll error values in Fig. 19 are unacceptable. Regarding the yaw angle the onboard estimates are more realistic and the estimation errors are within the desired limits after the transient.

Finally, the flight path angle values were obtained both for the GPS SBAS reference (γ^d) and the estimated data (γ) using a moving window technique with line fitting over the *Along position-Altitude position* values using 100 samples for each fit corresponding approximately to a 2 s sampling interval. γ^d and γ can be calculated from the slope of the lines. The flight path angle estimation errors are converging to the limits and mostly reach the desired intervals at around $X = -200$ m well before the flare phase. Note that in this case both the designated and estimated flight path angles are inaccurate and so this comparison is uncertain. Having an ILS system would provide more precise references to check the estimates.

As a summary, it can be stated that the developed delayed-ESKF algorithm is capable to estimate aircraft runway-relative position, velocity and orientation together with the unknown runway width and sensor biases. The algorithm gives acceptable results considering real flight data with a low-cost IMU and on-board camera system having estimation errors well inside the industrial tolerance values.

8. Conclusion

This work proposed an IMU–Camera–Barometric sensor-based estimation method for aircraft navigation during final approach. The aim of the proposed algorithm is to estimate the runway-relative position, velocity and orientation of an aircraft from IMU data, barometric data and runway features detected by a monocular camera. The challenge was to estimate the unknown IMU biases and runway size at the same time which has been rarely treated in the related work in literature. The paper first proved that the system model is globally state observable in realistic conditions of aircraft final approach. Then delayed-ESKF was applied in order to handle the time delay of the image-based measurements.

The estimation filter was first evaluated with simulated ideal data (no bias, no noise) and then with non-ideal data (sensor bias and noise added). Both non-delayed and delayed image data are considered to demonstrate the effectiveness of the delayed-ESKF compared to the normal one. When using delayed image information unacceptable results of the normal ESKF became acceptable with the delayed-ESKF. The simulated data tests also verified that sensor biases are accurately estimated.

After the simulation-based testing showed reliable performance the estimators (ESKF and delayed-ESKF) were tested on real flight data collected onboard the K-50 test aircraft. The estimated states converge around the reference GPS SBAS values for the delayed-ESKF proving the real flight applicability of the method.

The precision of estimates is further evaluated considering tolerances provided by Airbus in the VISIOLAND project as industrial requirements. Cross track, altitude and yaw errors were well inside their tolerance ranges. The roll angle could not be accurately evaluated because of uncertain reference data. The flight path angles are only approximately calculated (from along position and altitude) and show convergence towards the tolerance range but with some outliers. Considering the fact that cross track, altitude and yaw angle errors are well acceptable the results are promising. Future work may include implementation on the K-50 system (or other test aircraft) and testing the estimator in real-time during final approach.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 690811 and the Japan New Energy and Industrial Technology Development Organization under grant agreement No. 062800, as a part of the EU/Japan joint research project entitled "Validation of Integrated Safety-enhanced Intelligent flight cONtrol (VISION)".

The research reported in this paper and carried out partially at BME has been supported by the NRDI Fund (TKP2020 IES, Grant No. BME-IE-MIFM) based on the charter of bolster issued by the NRDI Office under the auspices of the Ministry for Innovation and Technology.

Part of this research was supported by the Ministry of Innovation and Technology NRDI Office within the framework of the Autonomous Systems National Laboratory Program.

The work of P. Bauer was partially supported by the János Bolyai Research Scholarship of the Hungarian Academy of Sciences and the ÚNKP-20-5 New National Excellence Program of the Ministry for Innovation and Technology.

The work of T. Gróf was partially supported by the ÚNKP-20-2 New National Excellence Program of the Ministry for Innovation and Technology.

The authors gratefully acknowledge the work of the Editor and Reviewers, the review comments in the first round greatly contributed to clarify the message of the paper, in the second round to improve the quality of presentation and detail observability and delay handling issues and in the third round to improve English language presentation.

Appendix. Noise covariances for real flight data

The following covariance matrices are considered while running the estimator on real sensor data sets:

$$\mathbf{Q} = 0.1673 \cdot \langle \theta_p, \theta_a, \theta_\omega, \theta_{ba}, \theta_{b\omega}, \theta_W \rangle$$
$$\theta_p = \langle 0.01,\ 1,\ 1.02 \rangle$$
$$\theta_a = \langle 0.91,\ 0.1,\ 0.1 \rangle$$
$$\theta_\omega = \langle 0.1,\ 0.1,\ 0.1 \rangle$$
$$\theta_{ba} = \langle 35 \cdot 10^{-2},\ 35 \cdot 10^{-3},\ 35 \cdot 10^{-3} \rangle$$
$$\theta_{b\omega} = \langle 35 \cdot 10^{-3},\ 35 \cdot 10^{-3},\ 35 \cdot 10^{-3} \rangle$$
$$\theta_W = \langle 64 \cdot 10^{-4} \rangle$$
$$\mathbf{R} = \langle 9,\ 4,\ 9,\ 4,\ 9,\ 4,\ 100 \rangle$$
$$\mathbf{P}_0 = \langle 10,\ 0.1,\ 0.1,\ 1.1,\ 0.1,\ 0.1,\ 0.1,\ 0.1,\ 0.1,\ 10.8,\ 0.8,\ 0.8,\ 0.1,\ 0.1,\ 0.1,\ 6 \rangle$$

In these equations diagonal matrices are denoted with $\langle .. \rangle$. The $\mathbf{Q}$ and the $\mathbf{P}$ covariance matrix values were set by trial and error, but the $\mathbf{R}$ covariance matrix represents the known measurement noises in the system. The camera error variance is known as $(4-9)^2$ pixels^2. According to the measurements the barometric sensor has an error variance of around 9 meters^2 compared to the GPS SBAS altitude. However, the tuning phase clearly showed that if the estimator relies heavily on the barometric sensor measurements the runway width estimation becomes less accurate which corrupts the velocity and position values as well. This has led to the adjusted last term in the $\mathbf{R}$ covariance matrix.

References

Aguirre, L. A., Portes, L. L., & Letellier, C. (2018). Structural, dynamical and symbolic observability: From dynamical systems to networks. *PLOS ONE*, *13*(10), 1–21. http://dx.doi.org/10.1371/journal.pone.0206180.

Andert, F., & Mejias, L. (2015). Improving monocular SLAM with altimeter hints for fixed-wing aircraft navigation and emergency landing. In *2015 International conference on unmanned aircraft systems* (pp. 1008–1016). http://dx.doi.org/10.1109/ICUAS.2015.7152390.

Azinheira, J. R., & Rives, P. (2008). Image-based visual servoing for vanishing features and ground lines tracking: Application to a UAV automatic landing. *International Journal of Optomechatronics*, *2*(3), 275–295. http://dx.doi.org/10.1080/15599610802303314.

Bianco-Martinez, E., Baptista, M. S., & Letellier, C. (2015). Symbolic computations of nonlinear observability. *Physical Review E*, *91*, Article 062912. http://dx.doi.org/10.1103/PhysRevE.91.062912, URL https://link.aps.org/doi/10.1103/PhysRevE.91.062912.

Burlion, L., & Kolmanovsky, I. (2020). Aircraft vision-based landing using robust extended command governors. In *Proc. of IFAC world congress 2020*. Berlin, Germany.

Conte, G., & Doherty, P. (2009). Vision-based unmanned aerial vehicle navigation using geo-referenced information. *EURASIP Journal on Advances in Signal Processing*, *2009*(1), http://dx.doi.org/10.1155/2009/387308.

Gibert, V., Plestan, F., Burlion, L., Boada-Bauxell, J., & Chriette, A. (2018). Visual estimation of deviations for the civil aircraft landing. *Control Engineering Practice*, *75*, 17–25. http://dx.doi.org/10.1016/j.conengprac.2018.03.004.

Gibert, V., & Puyou, G. (2013). Landing of an airliner using image based visual servoing. In *Proc. of 9th IFAC symposium on nonlinear control systems*.

Goupil, P., Boada-Bauxell, J., Marcos, A., Rosa, P., Kerr, M., & Dalbies, L. (2015). An overview of the FP7 reconfigure project: industrial, scientific and technological objectives. In *Proc. of the 9th IFAC symposium on fault detection, supervision and safety for technical processes* (pp. 976–981).

Grof, T., & Bauer, P. (2021). Voting based navigation system fault detection for aircraft position information during final approach. In *2021 5th International conference on control and fault-tolerant systems* (pp. 231–236). http://dx.doi.org/10.1109/SysTol52990.2021.9595390.

Grof, T., Bauer, P., Hiba, A., Gati, A., Zarandy, A., & Vanek, B. (2019). Runway relative positioning of aircraft with IMU-camera data fusion. In *Proc. of 21st IFAC symposium on automatic control in aerospace*.

Guilhamu, M. H. (2019). Airbus fleet readiness for GBAS/SBAS. URL https://www.icao.int/APAC/Meetings/Pages/2019-GBAS-SBAS-.aspx.

Hiba, A., Gati, A., & Manecy, A. (2021). Optical navigation sensor for runway relative positioning of aircraft during final approach. *Sensors*, *21*(6), http://dx.doi.org/10.3390/s21062203, URL https://www.mdpi.com/1424-8220/21/6/2203.

Hiba, A., Zsedrovits, T., Heri, O., & Zarandy, A. (2018). Runway detection for UAV landing system. In *CNNA 2018; the 16th international workshop on cellular nanoscale networks and their applications* (pp. 1–4).

Huang, L., Song, J., & Zhang, C. (2017). Observability analysis and filter design for a vision inertial absolute navigation system for UAV using landmarks. *Optik*, *149*, 455–468. http://dx.doi.org/10.1016/j.ijleo.2017.09.060.

Joo, S., Ippolito, C., Al-Ali, K., & Yeh, Y.-H. (2008). Vision aided inertial navigation with measurement delay for fixed-wing unmanned aerial vehicle landing. In *2008 IEEE aerospace conference*. http://dx.doi.org/10.1109/AERO.2008.4526557.

Larsen, T. D., Andersen, N. A., & Ravn, O. (1998). Incorporation of time delayed measurements in a discrete-time Kalman filter. In *Proc. of the 37th IEEE conference on decision and control*.

Le Bras, F., Hamel, T., Barat, C., & Mahony, R. (2009). Nonlinear image-based visual servo controller for automatic landing guidance of a fixed-wing aircraft.. In *Proc. of European control conference* (pp. 1836–1841).

Letellier, C., & Aguirre, L. A. (2009). Symbolic observability coefficients for univariate and multivariate analysis. *Physical Review E*, *79*, Article 066210. http://dx.doi.org/10.1103/PhysRevE.79.066210, URL https://link.aps.org/doi/10.1103/PhysRevE.79.066210.

Letellier, C., Sendiña-Nadal, I., & Aguirre, L. A. (2018). Nonlinear graph-based theory for dynamical network observability. *Physical Review E*, *98*, Article 020303. http://dx.doi.org/10.1103/PhysRevE.98.020303, URL https://link.aps.org/doi/10.1103/PhysRevE.98.020303.

Lynen, S., Achtelik, M., Weiss, S., Chli, M., & Siegwart, R. (2013). A robust and modular multi-sensor fusion approach applied to mav navigation. In *Proceedings of the ... IEEE/RSJ international conference on intelligent robots and systems. IEEE/RSJ international conference on intelligent robots and systems* (pp. 3923–3929). http://dx.doi.org/10.1109/IROS.2013.6696917.

Martinelli, A. (2011). *Closed-form solutions for attitude, speed, absolute scale and bias determination by fusing vision and inertial measurements*: *Research Report RR-7530*, INRIA, URL https://hal.inria.fr/inria-00569083.

Montanari, A. N., & Aguirre, L. A. (2020). Observability of network systems: A critical review of recent results. *Journal of Control, Automation and Electrical Systems*, *31*(6), 1348–1374. http://dx.doi.org/10.1007/s40313-020-00633-5.

Sola, J. (2017). *Quaternion kinematics for the error-state Kalman filter*: *Technical Report*, arXiv:1711.02508.

Strydom, R., Thurrowgood, S., & Srinivasan, M. V. (2014). Visual odometry: Autonomous UAV navigation using optic flow and stereo. In *Proc. of ICRA 2014*.

Tessier, Q., Macabiau, C., Milner, C., Azoulai, L., & Fernandez, F. (2017). Modelling the range and position error after EGNOS orbit and clock corrections. In *Proc. of the 30th international technical meeting of the satellite division of the institute of navigation*. http://dx.doi.org/10.33012/2017.15358.

Vidyasagar, M. (1993). *Nonlinear systems analysis*. Prentice Hall.

VISION (2016). VISION project webpage. URL https://w3.onera.fr/h2020_vision/node/1.

Watanabe, Y., Manecy, A., Amiez, A., Aoki, S., & Nagai, S. (2020). Fault-tolerant final approach navigation for a fixed-wing UAV by using long-range stereo camera system. In *2020 International conference on unmanned aircraft systems* (pp. 1065–1074).

Watanabe, Y., Manecy, A., Hiba, A., Nagai, S., & Aoki, S. (2019). Vision-integrated navigation system for aircraft final approach in case of GNSS/SBAS or ILS failures. In *Proc. of AIAA scitech 2019*. American Institute of Aeronautics and Astronautics, http://dx.doi.org/10.2514/6.2019-0113.

Weiss, S., Achtelik, M., Lynen, S., Achtelik, M., Kneip, L., Chli, M., & Siegwart, R. (2013). Monocular vision for long-term micro aerial vehicle state estimation: A compendium. *Journal of Field Robotics*, *30*, 803–831.

Zhang, L., Zhai, Z., He, L., Wen, P., & Niu, W. (2019). Infrared-inertial navigation for commercial aircraft precision landing in low visibility and GPS-denied environments. *Sensors*, *19*(2), http://dx.doi.org/10.3390/s19020408, URL https://www.mdpi.com/1424-8220/19/2/408.